Research Journal of Applied Sciences, Engineering and Technology 6(5): 807-811, 2013
DOI:10.19026/rjaset.6.4124
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: September 13, 2012 Accepted: October 24, 2012 Published: June 25, 2013

Research Article

Adaptive Ant Colony Algorithm for the VRP Solution of Logistics Distribution

Yu-Ping Wang
College of Automation, Beijing Union University, China

Abstract: In order to conquer the premature convergence problem and lower the cost of computing of the basic Ant Colony Algorithm (ACA), we present an adaptive ant colony algorithm, named AACA, coupled with a Pareto Local Search (PLS) algorithm and apply to the Vehicle Routing Problem (VRP) and Capacitated VRP (CVRP). By using the information entropy, the algorithm adjusts the pheromone updating strategy adaptively. Experiments on various aspects of the algorithm and computational results for some benchmark problems are reported. We compare our approach with some classic, powerful meta-heuristics and show that the proposed approach can obtain the better quality of the solutions.

Keywords: Ant colony algorithm, information entropy, pare to local search, vehicle routing problem

INTRODUCTION

In logistics distribution, the distribution path planning is main reason for the total operating costs. The optimization of Vehicle Routing Problem (VRP) in the logistics distribution is a well-known research widely concerned problem. Companies more and more attach importance to better design and manage their logistics distribution in order to meet higher level quality services at the lowest possible cost effort.

The VRP is a well-known combinatorial optimization problem with considerable economic significance. The main objective of VRP is to minimize the total required fleet size for serving all customers. Secondary objectives are to minimize the total distance traveled or to minimize the total route duration of all vehicles. A typical VRP can be described as the problem of designing least cost routes from one depot to a set of geographically scattered points, such as (cities, warehouses, customers, etc,), with the least effort. The VRP has been largely studied extensively because of the interest in its applications in logistic and supply-chains management.

As a well-known and complex combinatorial problem, the VRP has been largely study because of the interest in its applications in logistic and supply-chains management. In the early ’90s enterprise resource planning software vendors started to integrate tools to solve the VRP in supply chain management software (Aksoy and Derbez, 2003).

Ant Colony Algorithms (ACA), introduced by Colorni *et al*. (1991), Dorigo (1992) and Dorigo *et al*. (1996) is a population-based approach which was inspired by the observation of the behavior of ant colonies. The Ant System (AS) is a new distributed meta-heuristic for hard combinatorial optimization problems and was first used on the well known Traveling Salesman Problem (TSP). Starting from Ant System, several improvements of the basic algorithm has been proposed (Dorigo and Gambardella, 1997; Hu *et al*., 2008).

Dorigo *et al*. (1991) developed ant colony optimization (ACO) for the TSP. It performs better in problems such as the quadratic assignment problem (Maniezzo and Colorni, 1999), job shop-scheduling problem (Colorni *et al*., 1994) and the vehicle-routing problem (Bullnheimer *et al*., 1998; Mohan and Baskaran, 2012). Recently, Rizzoli *et al*. (2007) have started to use ACO algorithms for real-world applications. Angus and Woodward (2009) and López-Ibáñez and Stützle (2012) proposed an extension of ACO algorithms to tackle multi-objective combinatorial optimization problems. In all, many different versions of this problem have been formulated to take into account many possible different aspects.

This study proposes an adaptive ant colony algorithm based on Pareto local search (PLS) algorithm and information entropy to solve the premature convergence problem of the basic ant colony algorithm. By using the entropy, the algorithm adjusts the pheromone updating strategy adaptively for the VRP solution of logistics distribution.

ANT COLONY OPTIMIZATION

In this section, we first introduce the ACO. Then, we introduce the basic principles of ant colony optimization and we briefly present its application to the solution of the VRP and CVRP.

Vehicle routing problem: The vehicle routing problem is a very complicated combinatorial optimization

```
Initialize
  Set the size of the ant's population and put each ant
in the starting node
   do
     while (There are some ants which have not already
built its solution)
          Choose an ant which has not finished its trip
          The ant applies a state transition rule to
          incrementally build a solution
          Update the pheromone by local updating rule
   End
   Update the pheromone by global updating rule
   While (Don't conform any end conditions)
     Output the best solution
  End
```

Fig. 1: The ACS algorithm for an optimization problem

problem that has been worked on since the late fifties, because of its central meaning in distribution management.
The vehicle routing problem can be described as follows (Montemanni *et al*., 2003): n customers must beserved from a (unique) depot. Each customer i ask for a quantity q_i of goods. A fleet of v vehicles, each vehicle a with a capacity Q_a, is available to deliver goods. A service time s_i is associated with each customer. It represents the time required to service him/her. Therefore, a VRP solution is a collection of tours.

The VRP can be modeled in mathematical terms through a complete weighted digraph $G = (V, A)$, where $V = \{0,1,...,n\}$ is a set of nodes representing the depot (0) and the customers $(1,...,n)$ and $A = \{(i, j)|i,j\in V\}$ is a set of arcs, each one with a minimum travel time tt_{ij} associated. The quantity of goods q_i requested by each customer i ($i > 0$) am associated with the corresponding vertex with a label. Labels $Q_1,...,Q_v$, corresponding to vehicles capacities, are finally associated with vertex 0 (the depot). The goal is to find a feasible set of tours with the minimum total travel time. A set of tours is feasible if each node is visited exactly once (i.e. it is included into exactly one tour), each tour a starts and ends at the depot (vertex 0) and the sum of the quantities associated with the vertices contained in it, never exceeds the corresponding vehicle capacity Q_a.

The Capacitated Vehicle Routing Problem (CVRP) is the basic version of the VRP. The name derives from the constraint of having vehicles with limited capacity. The CVRP is NP-hard (Lenstr and Rinnooy Kan, 1981), since it contains one or more TSP as sub-problems. Obviously, a CVRP is more difficult to solve than a TSP.

Ant colony system: The ACS proposed by Dorigo and Gambardella (1997) is an algorithm for finding solutions to optimization problems and is shown in Fig. 1.

To solve the VRP, the artificial ants construct vehicle routes by successively choosing cities to visit, until each city has been visited. Whenever the choice of another city would lead to an infeasible solution for reasons of vehicle capacity or total route length, the depot is chosen and a new tour is started.

This heuristic uses a population of m agents which construct solutions step by step. When all the ants have constructed their tour, the best solution is rewarded so as to encourage the identification of ever better solutions in the next cycles.

Construction of vehicle routes: This process is responsible for the construction of new solutions. This is achieved using probabilistic stepwise solution construction. ACS goal is to find a shortest tour. In ACS m ants build tours in parallel, where m is a parameter. Each ant is randomly assigned to a starting node and has to build a solution, that is, a complete tour. A tour is built node by node: each ant iteratively adds new nodes until all nodes have been visited. When ant k is located in node i, it chooses the next node j probabilistically in the set of feasible nodes N_i^k (i.e., the set of nodes that still have to be visited). The probabilistic rule used to construct a tour is the following: with probability q_0 a node with the highest $[T_{ij}]^\alpha [n_{ij}]^\beta$, $j \ \varepsilon \ N^i_k$ is chosen, while with probability $(1-q_0)$ the node j is chosen with a probability p_{ij} proportional to $[T_{ij}]^\alpha [n_{ij}]^\beta$, $j \ \varepsilon \ N^i_k$.

With $\Omega = \{v_j \in V | v_j \text{ is feasible to be visited}\} \cup \{v_0\}$, city v_j is selected to be visited after city v_i according to a *random-proportional rule* (Dorigo and Gambardella, 1997) that can be stated as follows:

$$p_{ij} = \begin{cases} \dfrac{[\tau_{ij}]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{h\in\Omega} [\tau_{ij}]^\alpha \cdot [\eta_{ij}]^\beta} & \text{if } v_j \in \Omega \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Pheromone trail update: Once solutions have been evaluated, they can influence the pheromone matrix through a pheromone update process. After an artificial ant k has constructed a feasible solution, the pheromone trails are laid depending on the objective value L_k. For each arc (v_i, v_j) that was used by ant k, the pheromone trail is increased by $\Delta\tau_{ij}^k = 1/L_k$. In addition to that, all arcs belonging to the so far best solution (objective value L^*) are emphasized as if σ ants, so-called *elitist* ants had used them. One elitist ant increases the trail intensity by an amount $\Delta\tau_{ij}^*$ that is equal to $1/L^*$ if arc (v_i, v_j) belongs to the so far best solution and zero otherwise. Furthermore, part of the existing pheromone trails evaporates (ρ is the trail persistence) (Hu *et al*., 2008).

Thus, the trail intensities are updated according to the following:

$$\tau_{ij}^{new} = \rho \cdot \tau_{ij}^{old} + \sum_{k=1}^{m} \Delta\tau_{ij}^k + \sigma\Delta\tau_{ij}^* \tag{2}$$

where, m is the number of artificial ants.

Pheromone trail update: The most important component of an ant system is the management of pheromone trails. Ants accomplish task by depositing a pheromone as they move. This chemical trail can be detected by other ants, which are probabilistically more likely to follow a path rich in pheromone. In a standard ant system, pheromone trails are used in conjunction with the objective function for constructing a new solution.

Elitist strategy and other techniques can be utilized to improve algorithm stability and convergence. Initially no information is contained in the pheromone trail, meaning that all pheromone trails τ_{ij} are equal to a value τ_0. Since pheromone trails are updated by taking into account the absolute value of the solution obtained, τ_0 must take a value that depends on the value of the solutions that will be visited. Trail levels are updated after all the ants have constructed their solutions. The update is made according to the following equation:

$$\tau_{ij}(t+1) = \rho \cdot \tau_{ij}(t) + Q / f(X_i) \tag{3}$$

Where ρ is a coefficient which represents the trace's persistence; 1-ρ represents the evaporation and Q is a constant. $f(X_i)$ is value of objective function.

The update of the pheromone trail is done in a different way than those of the standard model where all the ants update the pheromone trail. Indeed, this manner of updating the pheromone trail implies a very slow convergence of the algorithm (Dorigo and Gambardella, 1997). For speeding-up the convergence, we update the pheromone trail by taking into account only the best solution produced by the search to date.

THE PROPOSED ALGORITHM

In this section, we present a hybrid algorithm. The approach applies pareto local search (PLS) algorithm to ACS.

Pareto local search: Pareto local search is an extension of local search algorithms for single objective problems. During the local search process, a solution s'∈ A can become dominated by recently introduced ones. If this is the case, such a solution s' is removed from the archive. In PLS, each solution in A has associated an additional visit-bit. The visit-bit is initialized to false and it is only set to true if all neighboring solutions of the solution associated to the visit-bit were already evaluated (Maniezzo and Colorni, 1999).

PLS terminates if all neighboring solutions of all solutions in A were explored, that is, every solution in A has the *visit-bit* set to *true*. However, PLS does not consider restrictions on the size of the archive and is a much more direct extension of usual iterative improvement algorithms.

Combining ACS with PLS for the VRP and CVRP: We applied PLS algorithms to the VRP and CVRP. PLS uses the weak component-wise ordering as an acceptance criterion. Once a complete solution is available, it is tentatively improved using a local search procedure. PLS algorithm iteratively selects a customer and tries to move it into another position within its zour or within another tour. A maximum computation time for the local search, t_{ls}, must be specified.

Pseudo-code of the ACS with PLS procedure for the vehicle routing problem is described in Fig. 2.

```
Best Cost := ∞ ;
τij = τ0   //For each arc ( i , j )
While (Not until computation time)
    For k:=1 to n
        Compute pij^k of n customers according to Eq. (1);
        Update the trail level τij^new according to Eq. (3);
  End−for
   Run PLS (maximum computation time = tls );
    Cost:= Cost of the current solution;
    If (Cost < Cost Best)
        Cost Best := Cost ;
        Best Sol := current solution ;
    End If
    For each move (i , j ) in solution Best Sol
        Update the trail level τij^new  according to Eq. (3);
   End For
End While
```

Fig. 2: Modified pseudo-code of AACA

The entropy-based hybrid ACS: Entropy comes from physics. It is used to describe chaos and disorder. The bigger the entropy value is , the more the confusion degree is . In the information theory, Shannon defined the information entropy as the probability of random event. A larger uncertainty of the variables has more information entropy. Chen *et al*. (2010) developed a method for multiple attribute decision making with interval-valued intuitionistic fuzzy information and applied to a practical firepower disposition problem.

In our approach, the information entropy is used to control the path selection and the pheromone updating strategy. In AACA, the path selection is related to the pheromone of each edge, which uncertainty exists. So we introduce the entropy to measure the uncertainty of pheromone in each edge and use information entropy to adjust the path chosen strategies and pheromone updated strategy. Specifically, for every *i*-th customer, i∈[I, m] of an ant during the process of constructing a solution we computed the entropy

$$S(t) = -k\sum_{j\in D} p_{ij}(t) \log p_{ij}(t) \tag{4}$$

where, p_{ij} is defined in Eq. (1).
We define:

$$\omega = 1 - k_1 \frac{S_{max} - S(t)}{S_{max}} \tag{5}$$

where, k_1 is positive constant, ω the proportion of pheromone update. We can get the information entropy value S (t) and determine the degree of uncertainty to choose the path. This definition is a combination of its own characteristics of the ACS algorithm, which combined a sequence of arithmetic with information entropy, to regulate the algorithm adaptively.

SIMULATION RESULTS

In this section we will present numerical results for AACA and compare them with results from previous methods ACS algorithm (Bullnheimer *et al*., 1997), TABUROUTE (TS) algorithm (Gendreau *et al*., 1994) for the VRP in Table 1. The numerical analysis was performed on a set of bench-mark problems described in Christofides *et al*. (1979).

Experiments were run on a Pentium IV, 2GB of RAM, 2.6 GHz processor. In order to assess the relative performance of ACS, TS versus AACA independently from the details of the settings, we choose the same settings. We used n artificial ants, initially placed at the customers v_i, ... , v_n and set $\alpha = 1$, $\beta = 5$ and $\rho = 0.75$. For all problems maximum iteration times are $2*n$.

Each run is guaranteed to be independent of others by starting with different random seeds. The result in Table 1 indicates that AACA was able to find good results for larger problem instances. AACA is superior to ACS except for four instances (C2, C6, C8 and C12). For the instances of C9 and C11, AACA even shows a slightly better performance than TS.

In order to verify further the effectiveness of AACA, 12 instances of CVRP benchmark problems are selected from Augerat Set A (instances A32k5, A54k7, A60k9, A69k9 and A80k10), Augerat Set B (instances B57k7, B63k10 and B78k10) and Christofides and Eilon (instances E76k7, E76k8, E76k10 and E76k14). These include the best-known solutions to each problem. These problems range from 32 customers to 80 customers and from 5 vehicles to 14 vehicles for the solution. For each instance of the datasets, the number of customers is given by the first number on the instance name. The main difference between these sets of problems is their tightness (the ratio between demand and capacity) and the location of customers. Solutions are then averaged for each problem type and the result is reported in Table 1. We used n=15 artificial ants and set α= 1, $q0$ = 0.8, β = 2 and ρ= 0.1. For all problems maximum iteration times are m = 30.

It is noted that the parameters of ACS is set as the proposed algorithm. Furthermore, we stop these algorithms after m = 30 continuous iterations if no improved solutions are found.

The simulation results are listed in Table 2. The table shows the best solution found by the proposed algorithm as well as the averages of the best solution found in each of the 30 runs. The column Optimum indicates the best known solution when our research started. The results reveal that AACA was able to find

Table 1: Experimental result for TS, ACS and PACS on VRP

Ins.	TS	ACS	AACA	AACA $_{dev}$ (%)
C1	524.61	524.61	524.61	0.00
C2	835.77	870.58	872.51	4.23
C3	829.45	879.43	868.4	6.45
C4	1036.16	1147.41	1044.1	11.57
C5	1322.65	1376	1361.6	4.09
C6	555.43	562.93	568.36	1.35
C7	913.23	948.16	946.07	4.23
C8	865.94	886.17	889.54	2.34
C9	1177.76	1202.01	1169.9	3.39
C10	1418.51	1504.79	1497.7	8.80
C11	1073.47	1072.45	1065.2	4.91
C12	819.56	819.96	827.6	1.05
C13	1573.81	1590.52	1584.7	3.20
C14	866.37	869.86	867.81	0.40

Table 2: Comparisons between AACA and ACS for 12 instances

Problem	Optimum	AACA	ACS
A32k5	784	786	792
A54k7	1167	1167	1180
A60k9	1358	1368	1375
A69k9	1167	1167	1167
A80k10	1764	1786	1813
B57k7	1153	1159	1163
B63k10	1496	1517	1534
B78k10	1266	1266	1275
E76k7	682	689	707
E76k8	735	735	735
E76k10	832	848	861
E76k14	1032	1039	1043

the better solutions then ACS for all instances. An interesting point is that AACA was able to find best solutions (instances A54k7, A69k9, B78k10 and E76k8).

CONCLUSION

Logistic distribution systems ask for high performance optimization algorithms. ACA algorithm proves to be among the best and most performing to vehicle routing. In this study, ant colony algorithm based on information entropy and local search is proposed for solving the VRP and the CVRP. Numerical results on some benchmark instances proved the efficiency of the proposed approach.

As future work, we intend to perform a detailed study on the importance of the method presented in this study, to conduct experiments to evaluate the effectiveness of AACA on real-world problems and wider range of combinatorial problems.

REFERENCES

Aksoy, Y. and A. Derbez, 2003. Software survey: Supply chain management. OR/MS Today, 30(3): 1-13.

Angus, D. and C. Woodward, 2009. Multiple objective ant colony optimization. Swarm Intell., 3(1): 69-85.

Bullnheimer, B., R.F. Hartl and C. Strauss, 1997. An Improved Ant System Algorithm for the Vehicle Routing Problem. Technical Report POM-10/97, Institute of Management Science, University of Vienna, Vienna, Austria.

Bullnheimer, B., R.F. Hartl and C. Strauss, 1998. Applying the Ant System to the Vehicle Routing Problem. In: Voß, S., S. Martello, I.H. Osman and C. Roucairol (Eds.), Meta-heuristics: Advances and Trends in Local Search Paradigms for Optimization. Kluwer, Boston, pp: 109-120.

Chen, Q., Z. Xu and S. Liu, 2010. A method based on interval-valued intuitionistic fuzzy entropy for multiple attribute decision making. Information, 13:67-77.

Christofides, N., A. Mingozzi and P. Toth, 1979. The Vehicle Routing Problem. In: Christofides, N., A. Mingozzi and P. Toth (Eds.), Combinatorial Optimization. Wiley, Chicester, pp: 315-338.

Colorni, A., M. Dorigo and V. Mariiezzo, 1991. Distributed optimization by ant colonies. Proceeding of Eearop. Conference Artificial Life. In: Varela, F. and P. Bourgine. (Eds.). Elsevier, Amsterdam.

Colorni, A., M. Dorigo, V. Maniezzo and M. Trubian, 1994. Ant system for job-shop scheduling. Belg. J. Oper. Res. Stat. Comput. Sci., 34: 39-53.

Dorigo, M., 1992. Optimization, learning and natural algorithms. Ph.D. Thesis, Polytechnic of Milan, Italy, (in Italian).

Dorigo, M. and L.M. Gambardella, 1997. Ant colony system: A cooperative learning approach to the traveling salesman problem. IEEE T. Evolut. Comput., 1(1): 53-66.

Dorigo, M., V. Maniezzo and A. Colorni, 1991. Positive feedback as a search strategy. Technical Report, University of Milan, Italy.

Dorigo, M., V. Maniezzo and A. Colorni, 1996. Ant system: Optimization by a colony of cooperating agents. IEEE T. Syst. Man Cyb., 26(1): 29-41.

Gendreau, M., A. Hertz and G. Laporte, 1994. A tabu search heuristic for the vehicle routing problem. Manage. Sci., 40: 1276-1290.

Hu, X., J. Zhang and Y. Li, 2008. Flexible Protein Folding by Ant Colony Optimization. In: Computational Intelligence in Biomedicine and Bioinformatics: Current Trends and Applications. Springer-Verlag, New York, pp: 317- 336.

Lenstr, J.K. and A.H.G. Rinnooy Kan, 1981. Complexity of vehicle routing and scheduling problems. Networks, 11: 221-227.

López-Ibáñez, M. and T. Stützle, 2012. An experimental analysis of design choices of multi-objective ant colony optimization algorithms. Swarm Intell., 6(3): 207-232.

Maniezzo, V. and A. Colorni, 1999. The ant system applied to the quadratic assignment problem. IEEE T. Knowl. Data En., 11(5): 769-778.

Mohan, B.C. and R. Baskaran, 2012. A survey: Ant colony optimization based recent research and implementation on several engineering domain. Expert Syst. Appl., 39: 4618-4627.

Montemanni, R., L. Gambardella, A. Rizzoli and A. Donati, 2003. A new algorithm for a dynamic vehicle routing problem based on ant colony system. 2nd International Workshop on Freight Transportation and Logistics.

Rizzoli, A.E., R. Montemanni, E. Lucibello and L.M. Gambardella, 2007. Ant colony optimization for real-world vehicle routing problems: From theory to applications. Swarm Intell., 1(2): 135-151.